TITLE OF THE INVENTION

# METHOD FOR DETERMINING LOSS RATE OR A LOSS CURVE OF A VIDEO SOURCE OR MULTIMEDIA TRAFFIC

## CROSS-REFERENCE TO RELATED APPLICATIONS

This application claims priority from U.S. provisional application serial number 60/171,669 filed on December 20, 1999 and incorporated herein by reference.

## STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR DEVELOPMENT

This invention was made with Government support under Grant No. MIP-9257103, awarded by the National Science Foundation. The Government has certain rights in this invention.

## REFERENCE TO A MICROFICHE APPENDIX

Not Applicable

## BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention pertains generally to the characterization of multimedia traffic sources and more particularly to a method of determining loss curves for multimedia traffic sources utilized within a packet network environment.

2. <u>Description of the Background Art</u>

The explosive growth of the Internet has spawned a number of video-based services which operate over packet networks, such as streaming video and video-on-demand. These applications often require, or can benefit from, an ability to provide Quality-of-Service (QoS) guarantees for the network. The QoS guarantees are usually in the form of bandwidth, end-to-end delay, and/or the loss rate experienced by the traffic stream. The rate variability of these video sources has introduced the need for characterizing the associated traffic so that resources within the network, such as bandwidth and buffer space, may be properly allocated during the call admission control (CAC) process. Traffic characterization also allows for the efficient policing of the traffic sources. Two of the primary resources allocated within the network are the transmission rate $\rho$ and the buffer size $B$. In an application where no losses are allowed, the video source can be characterized completely by determining the minimum buffer size necessary to avoid losses as a function of the rate $\rho$, and is referred to as the *burstiness curve*.

If the application can tolerate some amount of loss, the amount of bandwidth and/or buffer space required within the network can often be reduced significantly, since the burstiness curve of the source typically exhibits a long tail. The problem of determining the amount of network resources to allocate then becomes the problem of choosing a specific vector from the three-dimensional space $(\rho, B, \varepsilon)$. In order to

simplify the problem, either the transmission rate $\rho$ or the buffer size $B$ can be fixed and the corresponding curves calculated. The $\varepsilon$ versus $B$ curve for a specific video source transmission rate $\rho$ enables the estimation of the loss rate resulting from a given buffer size. A plot of the loss rate as a function of the buffer size for a given rate $\rho$ is referred to as the *loss curve* of the associated source at the given rate $\rho$.

## BRIEF SUMMARY OF THE INVENTION

The present invention provides efficient methods for determining and quantifying the effect of the variability of multimedia traffic sources into the loss characteristics of a multimedia traffic source. The *loss curve* of a multimedia traffic source characterizes the loss rate generated by the source as a function of the allocated buffer size for a given transmission rate. The loss curve is useful in the optimal allocation of resources when the traffic source is transmitted over a packet network, so that allocation tradeoffs between loss rate, bandwidth, and buffer space may be intelligently determined within the network. A deterministic method is taught which can provide an exact computation of the entire loss curve of traffic sources, such as elementary video streams, and MPEG-2 transport streams. The method exploits the piecewise linearity of the loss curve and computes only the points at which the slope of the loss curve changes, therefore, the method is capable of exactly characterizing the loss curve with the minimum number of points. The described methods have a small memory requirement while providing rapid execution times, for example, the entire loss curve of a two-hour elementary video stream was determined by an embodiment of present method in approximately 11 seconds on a Sun Ultra-2™ workstation. The efficiency of the

methods taught within the present invention are suitable for both off-line and on-line QoS provisioning in networked multimedia environments.

An object of the invention is to provide an efficient method for determining an exact loss curve for a traffic source.

Another object of the invention is to provide a method specifically tailored to the determination of loss curves for use with both elementary video streams and MPEG-2 transport streams.

Another object of the invention is to provide a method which exploits the piecewise linearity of the loss curves so as to reduce the necessary computational resources.

Another object of the invention is to provide a method in which the minimum number of subject points need be identified during computation for the level of accuracy desired.

Another object of the invention is to provide a method of determining loss curves which is computationally space-efficient.

Further objects and advantages of the invention will be brought out in the following portions of the specification, wherein the detailed description is for the purpose of fully disclosing preferred embodiments of the invention without placing limitations thereon.

## BRIEF DESCRIPTION OF THE DRAWINGS

The invention will be more fully understood by reference to the following drawings which are for illustrative purposes only:

FIG. 1 is a schematic of a data-flow model utilized for determining the loss curves of a traffic source according to an embodiment of the present invention.

FIG. 2 is pseudocode for determining active periods for a traffic stream according to an embodiment of the present invention.

FIG. 3 is a graph of busy periods within an exemplified portion of a traffic stream for various values of buffer size $B$ according to an aspect of the present invention.

FIG. 4 is a graph of active periods within an exemplified portion of a traffic stream along with associated queue sizes according to an aspect of the present invention.

FIG. 5 is pseudocode for determining the loss curve for a traffic stream according to an embodiment of the present invention.

FIG. 6 is a graph of transport rate within an exemplified portion of an MPEG-2 transport stream according to an aspect of the present invention.

FIG. 7 is a graph of busy periods within an MPEG-2 transport stream according to an aspect of the present invention.

FIG. 8 is a graph of the loss curve for a "Lambs" trace determined according to an embodiment of the present invention.

FIG. 9 is a graph of $B$ versus $\rho$ for the "Lambs" trace shown in FIG. 8.

## DETAILED DESCRIPTION OF THE INVENTION

Referring more specifically to the drawings, for illustrative purposes the present invention is embodied in the apparatus generally shown in FIG. 1 through FIG. 9. It will be appreciated that the apparatus may vary as to configuration and as to details of the

parts, and that the method may vary as to the specific steps and sequence, without departing from the basic concepts as disclosed herein.

Traffic streams are increasingly being communicated over packet networks such as the Internet. One form of multimedia that is commonly communicated over the Internet is video which is typically transported over a packet network as an elementary video stream consisting of a sequence of frames generated at a fixed rate (frame period), which may be of varying sizes due to scene changes. An embodiment of the methods according to the present invention is presented for providing an exact determination of loss and loss curves for an elementary video stream.

FIG. 1 shows a data-flow model 10 utilized for computing the loss curve for an elementary video stream. The traffic stream 12 is sequenced through a peak-rate shaper 14 comprising an input queue mechanism 16 within which a new time sequence for the bit-stream of the input traffic is created. A small sequence of frames 18a-18g is presently shown represented within the queue 16. In response to different values of the peak rate $r$, the maximum queue depth 20, denoted by $s(r)$, within input queue 16 of peak-rate shaper 14 will vary. Frames are removed from input queue 16 by an output 22 of the traffic stream at a rate $r$. The bit-stream 24 at the output of peak-rate shaper 14, denoted by $a(r,t)$ is also dependent on the peak rate $r$. The present invention provides a method for computing the loss curve of the traffic source at the output of the peak-rate shaper 14 for a specific transmission rate $\rho$ and buffer size $B$.

According to the model 10, the traffic stream output 24 $a(r,t)$ then enters a secondary shaper 26, having a secondary queue 28, with a maximum queue length 30 wherein frames are output at a deterministic output rate 32, denoted by $\rho$, for a given buffer size $B$. By recording the amount of data lost from this second buffer in response to different buffer sizes, the loss curves for the source can be obtained for the given peak rate $r$ and transmission rate $\rho$. The $B$ versus $\rho$ curve for a given maximum acceptable loss rate $\varepsilon$ is then obtained by constructing a series of loss curves for different choices of the rate $\rho$ and reading off the associated $B$ values corresponding to the given loss rate.

The amount of video traffic within the video stream which is lost over the time interval $[0,t]$ is defined as $L_{B,\rho},\ \rho(t)$. The fraction of lost traffic, or the loss rate $\varepsilon(B,\rho)$, is therefore given by:

$$\varepsilon(B,\rho)=\frac{L_{B,\rho}(T)}{M} \tag{1}$$

where $M$ indicates the total number of bits in the elementary stream with $T$ being the duration. To plot a complete loss curve for a given rate $\rho$, it is only necessary to consider the range of buffer sizes $0 \leq B \leq \sigma(\rho)$, where $\sigma(\rho)$ is the maximum burstiness of the source at rate $\rho$. For a complete characterization of the source, a series of loss curves can thereby be constructed for different transmission rates in the range $0<\rho<r$. To complete the definition of the loss curve, the loss rate for the boundary values of $B$ and $\rho$ is defined as follows:

$$\varepsilon(B,\rho)=0, \quad B>\sigma(\rho) \tag{2}$$

$$\varepsilon(0,\rho)=\frac{r-\rho}{r}, \quad 0\le\rho\le r \tag{3}$$

$$\varepsilon(B,0)=1, \quad 0\le B\le M \tag{4}$$

$$\varepsilon(B,0)=0, \quad B\ge M \tag{5}$$

To analyze the behavior of the queue at the second shaper in FIG. 1, the on-off signal at the output of the peak-rate shaper must first be characterized. The "on" periods of the signal are referred to as "active-periods" and a computation method is presented to determine these active periods for the bit-stream $a(r,t)$ at the output of the peak-rate shaper for the traffic source. Information about these active periods is then utilized in the computation of the loss curve.

The *active period* is defined as a maximal period of time during which the peak-rate shaper is continuously transmitting traffic, which corresponds to an on-period of the signal $a(r,t)$. Letting $n_a(r)$ denote the number of active periods of the signal for a peak rate of $r$, $s_i^r$ the time instant at which the *i*th active period commences, and $t_i^r$ the time when it ends. The active periods $(s_i^r, t_i^r)$, of the traffic stream for $1\le i\le n_a(r)$ are computed.

Assuming that the number of frames in the video trace is $N$, the length of the trace is $T$, the frame rate is $f$, and the size of the *i*th frame is $d_i$ bits, the maximum frame size in the trace can be defined as $d_{\max}=\max_{1\le i\le N} d_i$. In addition, it is assumed that a frame is added instantaneously to the queue of the peak-rate shaper at the end of the corresponding frame period, such that the first frame arrives in the queue at time

$1/f$, which marks the beginning of the first active period. When the peak rate $r$ satisfies $r \geq d_{\max}/f$, it becomes simple to compute the active periods of the signal $a(r,t)$.

$$s(r) = d_{\max}, \qquad n_a(r) = N, \qquad s_i^{\,r} = \frac{i}{f}, \quad \text{and} \quad t_i^{\,r} = s_i^{\,r} + \frac{d_i}{r} \tag{6}$$

However, in the general case when $r < d_{\max}/f$, neighboring frames overlap with each other in the shaper queue, leading to larger maximum queue lengths and a smaller number of active periods. Letting $q_i(r)$ represent the size of the queue at the input of the peak-rate shaper following the arrival of the *i*th frame. It will be appreciated that the maximum queue length always occurs immediately after the arrival of a frame, and is given by:

$$s(r) = \max\nolimits_{1 \leq i \leq N} q_i(r) \tag{7}$$

The active periods of the elementary stream can be determined by traversing the sequence of frames and computing the queue size at the instant subsequent to the arrival of each frame.

FIG. 2 is the pseudocode for determining the active periods for a given value of the peak rate $r$ and it processes the individual frames of the elementary stream in sequence and computes the maximum queue size $s(r)$, the number of active periods $n_a(r)$, and the starting and ending times of each active period $(s_i^{\,r}, t_i^{\,r})$, $1 \leq i \leq n_a(r)$.

Information about the active periods of the signal $a(r,t)$ can be utilized to compute the loss curve of the original video stream by observing the queue behavior at

the input of the second shaper 26 shown in FIG. 1. A method is described for determining the loss curve for a specific service rate $\rho$. For the sake of simplicity, the parameter $r$ is omitted from the notations in the remainder of the discussion as it remains constant.

The peak-rate shaping procedure produces a sequence of active periods for a given peak-rate $r$. If the output rate of the peak-rate shaper is denoted by $m(t)$, then:

$$m(t)=\begin{cases} r, & t\in[s_i,t_i] \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

The maximal period of time during which the queue of the second shaper in FIG. 1 remains non-empty is then defined as a *busy period*. The notation $\alpha_i, \beta_i$, is utilized to represent, respectively, the starting and ending times of busy period $i$. For a given sequence of active periods, the corresponding busy periods are a function of the transmission rate $\rho$ and the buffer size $B$ at the input of the second shaper 26.

FIG. 3A shows an example of eight active periods, 50 through 64, within a portion of an elementary video stream and it corresponds with FIG. 3B depicting three busy periods 66, 68, 70 associated with the active periods of FIG. 3A when no buffer size constraints have been imposed.

Assuming no losses from the buffer, the local maximum queue size is denoted by $Q_{B,i}^*$ for busy period $i$, a buffer size $B$, and a time $\tau_{B,i}$ at which the local maximum occurs. For simplicity, the subscript $B$ is omitted when the buffer size is obvious from

the context. The present method for characterizing the loss curve of the traffic source comprises two key observations:

1. for a given transmission rate $\rho$, the loss rate $\varepsilon$ is a *piecewise-linear* function of the buffer size $B$, such that as the buffer size is decreased, the slope of the loss curve can change only when,

   (i) a busy period starts to experience losses for the first time, or

   (ii) a busy period breaks into two or more constituent busy periods,

   thereby enabling the exact computation of the loss curve by identifying the points at which such events occur;

2. within each busy period where a loss occurs, the last time instant at which a loss occurs is the time instant $\tau_{B,j}$ at which the maximum queue size occurs.

The loss curve of the source, for a given transmission rate $\rho$, can be constructed by considering various values of the buffer size $B$ in the range $0 \leq B \leq \sigma(\rho)$, and computing the loss rate for each of these cases. This procedure, however, provides only an approximation of the actual loss rate between the points considered and it requires a simulation of the queue for each incremental buffer size. The present invention, by contrast, exploits the piecewise-linearity of the loss curve by computing the slope of the loss curve at positions where the slope changes. Therefore, a simulation of the queue is avoided because only the changes in the busy periods of the queue are tracked with a change in buffer size. As a result of limiting the points of interest to those

at which curve deflection may occur, the present method provides an efficient loss curve computation method capable of determining exact loss curves comprising a minimum number of points and associated computations.

A change in the number of busy periods experiencing losses constitutes the only case in which the slope of the loss curve may change as the buffer size $B$ is decreased towards its minimum value of zero. FIG. 3A shows the active periods 50 through 64 of the input signal which are mapped into FIG. 3B as three busy periods 66, 68, 70, generated for a model having an infinite buffer size. The global maximum queue size occurs in the second busy period 68 and is denoted by $Q_{\infty,2}^{*}$. As the buffer size $B$ is decreased, losses do not begin to occur until the queue size for this example drops to below the global maximum queue size $Q_{\infty,2}^{*}$. As the queue size is further decreased to a value $B_1 < Q_{\infty,2}^{*}$, as shown in FIG. 3C, losses occur in the second busy period 68. If the peak at time $t_1$ corresponding to the end of active period 58 is the only peak experiencing losses, the total amount of lost traffic is equal to $\left(Q_{\infty,2}^{*} - B\right)$, which is a linear function of $B$. It should also be appreciated that no losses can occur to the right of $t_1$ in the same busy period.

On further decreasing the buffer size to $B_2 < B_1$, as shown in FIG. 3D, the second busy period 68 splits into two busy periods 68a, and 68b, with the break occurring at time $t_2$. After the split, both resultant busy periods 68a, 68b, experience losses, wherein the total amount of lost data is equal to

$\left(Q_{B2,2}^{\bullet} - B\right) + \left(Q_{B2,3}^{\bullet} - B\right) = \left(Q_{B2,2}^{\bullet} + Q_{B2,3}^{\bullet}\right) - 2B$, where $Q_{B2,2}^{\bullet}$ and $Q_{B2,3}^{\bullet}$ are the corresponding local maximum queue sizes of the new busy periods 68a and 68b, respectively, as indicated in FIG. 3D. Therefore, the amount of lost data is again a linear function of $B$, and its slope is determined by the number of busy periods during which losses are experienced.

For a video stream of finite duration, the values of the buffer size $B$ that cause either a loss in a busy period with no prior loss, or a break in a busy period that already experiences loss, form a *finite* set. It will be appreciated that only the values of $B$ belonging to this set need be computed to exactly determine the loss curve, since the loss curve is piecewise linear between adjacent buffer points belonging to this finite set. Allowing $A(t_1,t_2)$ to denote arrivals into the buffer during the interval $[t_1,t_2]$, and the number of bits lost in the interval $[t_1,t_2]$ to be given by $L_{B,\rho}(t_1,t_2)$ when the buffer size is $B$ and the transmission rate is $\rho$. For simplicity, in the special case when $t_1 = 0$ and $t_2 = t$, the notation $L_{B,\rho}(t)$ is utilized instead of $L_{B,\rho}(t_1,t_2)$. For a given busy period in which losses are experienced, the amount of loss increases linearly with a slope of $(-1)$ as the buffer size $B$ is decreased, insofar as the busy period does not subdivide into multiple busy periods. In view of these observations, the piecewise linearity of the loss curve can be illustrated.

**Lemma 1** - Letting $\gamma_i$ represent the last instant at which a loss occurs within busy period $i$, the amount of data lost during the busy period is given by:

$$L_{B,\rho}(\alpha_i,\beta_i) = A(\alpha_i,\gamma_i) - \rho(\gamma_i - \alpha_i) - B \quad (9)$$

where $\alpha_i$ is the starting time of the busy period $i$.

**Proof** - The losses from the queue during the interval $(\alpha_i,\gamma_i)$ must be equal to the arrivals into the queue during the interval minus the total traffic transmitted during the interval, minus the bits remaining in the buffer at the end of the interval. Since the queue does not underflow during the interval $(\alpha_i,\gamma_i)$, the total traffic transmitted during the interval $(\alpha_i,\gamma_i)$ is $\rho(\gamma_i - \alpha_i)$. Furthermore, since losses occur at time $\gamma_i$, the buffer occupancy at time $\gamma_i$ is $B$. Subtracting these two terms from the bit arrivals results in the equation Eq. 9.

Therefore, in order to calculate the losses during the busy period, it is sufficient to determine the starting time of the busy period and the last instant $\gamma_i$ at which losses occur during the busy period. Note also that $\gamma_i$ must coincide with the end of an active period of the source. It will be subsequently shown that $\gamma_i$ coincides with the time instant at which the local maximum queue size would have occurred during the busy period if the buffer size were infinite. Lemma 1 may be utilized to show that the loss curve is piecewise linear.

**Lemma 2** - For a given transmission rate $\rho$, the loss curve of an elementary video stream is piecewise linear. The slope of the curve changes only at values of the buffer size $B$ where one of the following events occurs:

1. the number of busy periods changes within which losses occur;

2. the last instant at which a loss occurs $\gamma_i$ in a busy period $i$ shifts for any busy period $i$.

**Proof** - Consider two distinct values of $B$, $B_1$ and $B_2$, with $B_1 < B_2$, such that (i) the number of busy periods undergoing losses remains the same at $B_1$ and $B_2$; and (ii) the last instant at which a loss occurs in each of these busy periods, $\gamma_i$, also remains the same. Letting $S_l$ denote the set of busy periods in which losses occur, then according to Lemma 1 the total amount of lost data over the entire duration $T$ of the video stream is given by:

$$L_{B,\rho}(T) = \sum_{i \in S_l} \left( A(\alpha_i, \gamma_i) - \rho(\gamma_i - \alpha_i) \right) - n_l B, \quad B_1 \le B \le B_2 \tag{10}$$

where $n_l$ is the number of busy periods in the set $S_l$. As a result, the plot of $L_{B,\rho}(T)$ and therefore that of $\varepsilon(B,\rho)$ with respect to $B$ in the range $B_1 \le B \le B_2$ is a straight line segment with slope $-n_l$.

The entire loss curve of the elementary video stream for a given transmission rate $\rho$ may therefore be obtained by starting from a buffer size equal to the corresponding burstiness value $\sigma(\rho)$, which is equal to the global maximum queue size when the buffer size is infinity, and progressively considering successive buffer sizes at which either (i) a busy period with no prior loss starts to experience losses, or (ii) a busy period experiencing loss breaks into smaller busy periods. The time instant at which the last loss occurs within a given busy period is the time at which the queue size reaches its local maximum within the busy period, when no losses occur from the buffer. The

aforementioned procedure provides a simple method of determining the parameter $\gamma_i$ in Eq. 10 preceeding a loss rate computation.

The size of the buffer at the instant at which the first loss occurs during a busy period can easily be identified by computing the local maximum queue size within the busy period while ignoring any losses. The size of the buffer at the instant that a break occurs in a busy period, however, is more difficult to identify. The starting instants of active periods within a busy period are the points at which a break may potentially occur, because the queue size reaches local minimums at the start of active periods. Therefore, the maximum buffer size at which a break occurs within the busy period may be determined by computing the buffer size that causes the queue size to be zero at each of these points and selecting the maximum among all the points. This above procedure, however, is cumbersome because the effect of losses must be accumulated over multiple active periods to determine the buffer size at which the queue size reaches zero exactly at the start of a given active period. An enhanced method is thereby provided which identifies the buffer size at which a break in the busy period occurs.

FIG. 4 illustrates an example of the enhanced approach wherein seven active periods 72 through 84 result in a single busy period 86. The peaks and valleys of the queue depth within the busy period 86 correspond to the ending and starting times, respectively, of the active periods 72 through 84. The illustrated busy period is shown for an unlimited buffer size wherein no losses occur and the maximum depth of the queue occurs at time $t_6$, which corresponding to the end of the sixth active period 82.

As the buffer size is decreased from infinity, losses start to occur first during the sixth active period 82, which causes a corresponding dip in the valleys following the peak at $t_6$, resulting in a break in the busy period 86 if the queue size drops to zero at any of the valleys.

As buffer size decreases, the attendant losses start to occur progressively from the highest peak, to the next highest, and so on. In addition, if a loss occurs from one of the peaks within the busy period, no losses can occur from a following peak unless the latter is larger than the former, or the busy period breaks. For example, in FIG. 4 no losses can occur between the peaks at $t_1$ and $t_4$ without first causing a break in busy period 86. The present procedure utilizes this observation by processing only the active periods which correspond to monotonic queue size increases within the busy period wherein subject buffer sizes are identified at which a break may occur.

The method of identifying the buffer size at which a break may occur in the busy period involves constructing a sequence of active periods $S$ within a busy period having monotonically increasing queue sizes and including as part of the sequence the last active period. For busy period 86 exemplified in FIG. 4, the sequence consists of the active periods 72, 78, 82, 84. Starting from the end of the sequence, potential break points are then examined for a break in the busy period. For each active period within the sequence, the buffer size is identified at which the depth of the queue reaches zero within the interval between the current active period and the previous active period of the sequence. For instance, in relation to FIG. 4, the procedure first determines the buffer size at which the queue size will reach zero at the valley between $t_6$ and $t_7$,

which expresses the difference between the queue sizes at $t_6$ and the following valley. After reducing the buffer size, if the buffer size exceeds the next peak being considered in the sequence, the peak at $t_4$ as per this example, a break will result between $t_6$ and $t_7$ before any losses occur at time $t_4$, and no further processing of the sequence is necessary. If the peak at time $t_4$ is less than the identified buffer size, however, the procedure determines the concomitant buffer size associated with a break occurring between $t_4$ and $t_6$. The aforementioned process continues until a peak which exceeds the currently identified buffer size is located, or the sequence of active periods in $S$ is exhausted, wherein the buffer size is chosen as the last identified point at which the break occurs.

FIG. 5 illustrates high-level pseudocode for determining the loss curve according to the present invention and commences by computing busy periods when the buffer size $B$ is infinite. A function *process_active_periods()* processes the active periods *1* to $n_a$ in order to determine the set of busy periods, after which it records all buffer size points, and associated cause, at which any busy period starts experiencing loss. Each of these recorded buffer points correspond to a maximum queue size for each of the busy periods and the function computes and records the maximum buffer sizes at which each busy period breaks. In the iteration of step *2* within the pseudocode, the recorded maximum buffer points are retrieved and processed. Each of these buffer points is associated with a busy period $b$ along with an associated cause which comprises either of two cases, (1) LOSS - wherein a loss in the busy period $b$ occurred, and (2) BREAK -

wherein a break in the busy period occurred. For the LOSS case the cause is a loss in the busy period, and the procedure updates the loss variables ($L_B$ and $n_l$ variables) that are used in the computation of the total loss for the current value of the buffer size, and the total amount of loss is given by $(L_B - n_l B)$. In the case of a break cause, the function updates the loss variables again and then processes the break by calling a function *process_break()* which commences by computing new busy periods generated after the break of busy period $b$, which are each checked for possible losses. For each busy period within which a loss is experienced the function updates the set of loss variables. Furthermore, the function records the maximum buffer size associated with the new busy periods at which losses commence within the busy periods. The procedure then determines and records the buffer sizes associated with the first break within the set of all generated busy periods. To avoid the output of multiple points at the same buffer size value the function outputs a $(B, \varepsilon)$ pair only when a new buffer size is processed by calling the function *output_point()*. The aforesaid procedure repeats for each new maximum buffer point and associated busy period which has been recorded, and terminates after all recorded buffer points have been processed. The loss curve may then be plotted by interconnecting the loss ratio $(\varepsilon)$ values at each buffer point with linear segments.

The worst-case time and space complexities of the method may be determined by considering the computations performed at each of the requisite buffer points. The number of buffer points is given by $O(n_a)$, where $n_a$ is the number of active periods,

since a buffer point corresponds to either a break or a loss occurring in a busy period. The number of steps to process a busy period for a specific buffer size $B$ is also given by $O(n_a)$. In calculating the number of periods and steps, the time to compute any new busy periods and the buffer size values at which either a loss or a break occurs in each of these busy periods has been included. Therefore, the total worst-case time complexity for the execution of the method is given by $O(n_a^2)$. The space-complexity, as defined by the space required for storing the outputs are proportional to the number of buffer points processed by the method, and are therefore also given by $O(n_a)$.

The loss curve which has been determined by the method can be utilized for computation of a $B$ versus $\rho$ curve, for instance, a plot of the minimum buffer size $B$ in relation to the service rate $\rho$ for a given specific loss rate $\varepsilon$. In plotting a $B$ versus $\rho$ curve, a determination is made of the loss curve for every $\rho$ and the associated values for $B$ are then plotted for each specified value of $\varepsilon$. The aforementioned procedure may be performed with any desired level of granularity for the rate $\rho$, or alternatively performed at rate points associated with the burstiness curve for the stream.

The preceding example focused on the determination of loss curves within an elementary video stream, however, the present invention is applicable to any form of traffic stream within a packet based network. In a further example, the inventive methods are applied to an MPEG-2 transport stream. The MPEG-2 transport stream format comprises a grouping of one or more programs within a single stream, each *program* being defined as a grouping of elementary streams (audio, video, teletext, and

so forth) which have a common time-base for delivery. The MPEG-2 transport stream is currently a preferred choice for communicating multimedia streams within error-prone environments such as exist within today's packet-switched networks. A basic characteristic of the MPEG-2 transport stream format is that the throughout rate of the stream is piecewise constant, as a result, the MPEG-2 transport stream comprises a sequence of constant-rate segments which are exemplified within FIG. 6.

Referring to FIG. 6, it will be appreciated that the same definition for the busy period may be utilized with MPEG-2 transport streams, wherein a busy period commences at the beginning of a rate segment *i* at which $r_i > \rho$, wherein $r_i$ is given as the rate of the segment.

FIG. 7 exemplifies a portion of an MPEG-2 transport stream having rate segments $r_1$ through $r_{12}$ and an associated busy period 88. Homologous to the case of elementary video streams, the amount of data lost in a busy period is given by Eq. 8, where $m(t)$ corresponds to the transport rate of the MPEG-2 transport stream. Similarly, the total losses throughout the whole stream is given by Eq. 9. Therefore, in computing the exact loss curve in the transport stream case, only the buffer points need be considered at which either (i) a change occurs in the index of the last rate segment experiencing loss within a busy period, or (ii) a change occurs in the number of busy periods that experience a loss. Computation of the next buffer size at which a break occurs within a given busy period is performed differently for MPEG-2 transport streams than within elementary video streams. Computed first are the set of rate segments that provide increasing peaks of the queue size, these increasing peaks can occur at the

ends of any particular rate segment. Within a busy period, the time instants necessitating a check for a possible break comprise only those instants of time which correspond to valleys which exist within the busy period and do not include all time instants which correspond to the end of decreasing line segments within the busy period. As an example of determining possible breaks within a busy period consider that the rate segments $r_3$, $r_6$, and $r_9$ of FIG. 7, are inserted into the set of increasing queue peaks, whereupon it is sufficient to examine the valley points pursuant to rate segments $r_4$, $r_8$, and $r_{11}$ in determining buffer sizes associated with a first break in the busy period.

Traces were characterized for loss according to the present methods to determine the efficiency of the methods on real-world traffic communicated over the Internet. The traces were provided as elementary video streams, however, it will be appreciated that MPEG-2 transport streams, and other traffic stream formats are expected to show similar efficiency results. The present loss curve determination methods were applied to a number of elementary video stream traces each comprising a segment of approximately 30 minutes in duration, with the exception of the "Garrett's" trace which contained a segment of approximately 120 minutes duration. The long "Garrett's" trace was created by Mark Garrett for his thesis entitled "*Contributions Toward Real-Time Services On Packet-Switched Networks*", and consists of 174,136 frames with a frame rate or 24 Hz. Table 1 illustrates execution times for the method according to the preferred embodiment in determining loss curves applied to the associated traces which were obtained from a variety of video programming sources,

which included movie segments, such as “Mr. Bean”, and “Silence of the Lambs” in addition to television programming, such as network news programs and “The Simpsons”. It will be appreciated that the execution times, in toto, are relatively small for the present method, and that loss was determined for each of these traces within a few seconds when executed on a Sun Ultra-2™ workstation. The number of points on the loss curve varied from 9,632 for the “Lambs” trace up to 89,489 for the “Garrett's” trace, which was comparable to the total number of frames in the respective traces. FIG. 8 illustrates a loss curve and FIG. 9 illustrates a typical $B$ versus $\rho$ curve which were determined for the “Lambs” trace according to the present invention.

Accordingly, it will be seen that this invention provides an accurate and rapid method for determining loss and loss curves for traffic sources being communicated over packet networks. The described methods may be implemented within a variety of computers and communication related equipment which contain computational resources. The method for determining loss and loss curves was exemplified within the description for use with both elementary video streams as well as MPEG-2 transport streams. It will be appreciated that the methods described according to the present invention may be implemented for the computation of loss and loss curves for a variety of additional traffic sources, such as may be subject to transmission over a packet network.

Although the description above contains many specificities, these should not be construed as limiting the scope of the invention but as merely providing illustrations of some of the presently preferred embodiments of this invention. Thus the scope of this

invention should be determined by the appended claims and their legal equivalents. Therefore, it will be appreciated that the scope of the present invention fully encompasses other embodiments which may become obvious to those skilled in the art, and that the scope of the present invention is accordingly to be limited by nothing other than the appended claims, in which reference to an element in the singular is not intended to mean "one and only one" unless explicitly so stated, but rather "one or more." All structural, chemical, and functional equivalents to the elements of the above-described preferred embodiment that are known to those of ordinary skill in the art are expressly incorporated herein by reference and are intended to be encompassed by the present claims. Moreover, it is not necessary for a device or method to address each and every problem sought to be solved by the present invention, for it to be encompassed by the present claims. Furthermore, no element, component, or method step in the present disclosure is intended to be dedicated to the public regardless of whether the element, component, or method step is explicitly recited in the claims. No claim element herein is to be construed under the provisions of 35 U.S.C. 112, sixth paragraph, unless the element is expressly recited using the phrase "means for."

| Trace | Frame Rate (Hertz) | Number of Frames | Video Length (minutes) | Running Time (seconds) | Space (# points) |
|---|---|---|---|---|---|
| mr. bean | 25 | 40000 | 26.7 | 2.3 | 18812 |
| asterix | 25 | 40000 | 26.7 | 2.8 | 24345 |
| atp | 25 | 40000 | 26.7 | 2.5 | 22641 |
| bond | 25 | 40000 | 26.7 | 2.9 | 27169 |
| dino | 25 | 40000 | 26.7 | 1.9 | 14293 |
| lambs | 25 | 40000 | 26.7 | 1.6 | 9632 |
| movie2 | 25 | 40000 | 26.7 | 2.1 | 17634 |
| mtv1 | 25 | 40000 | 26.7 | 3.3 | 28262 |
| mtv2 | 25 | 40000 | 26.7 | 5.2 | 22525 |
| news1 | 25 | 31515 | 21.0 | 2.1 | 20036 |
| news2 | 25 | 40000 | 26.7 | 2.1 | 16888 |
| race | 25 | 40000 | 26.7 | 4.2 | 33317 |
| sbowl | 25 | 40000 | 26.7 | 2.5 | 24044 |
| simpsons | 25 | 40000 | 26.7 | 2.3 | 20527 |
| soccer1 | 25 | 40000 | 26.7 | 3.2 | 30265 |
| soccer2 | 25 | 40000 | 26.7 | 3.1 | 27129 |
| star | 25 | 40000 | 26.7 | 1.8 | 12189 |
| talk1 | 25 | 40000 | 26.7 | 1.9 | 14110 |
| talk2 | 25 | 40000 | 26.7 | 2.0 | 15683 |
| terminator | 25 | 40000 | 26.7 | 1.7 | 11396 |
| Garrett's trace | 24 | 174136 | 120.9 | 11.2 | 89489 |